scisys [1]

esa [2]

EAGLE: An Extensible, End-to-End Simulation and Evaluation Framework for Planetary E/DLS

Ender St. John-Olcayto (1), Guy Johns (2), Alastair Pidgeon (1), Christian Philippe (2)

Entry, Descent, and Landing: A Tough Engineering Challenge!

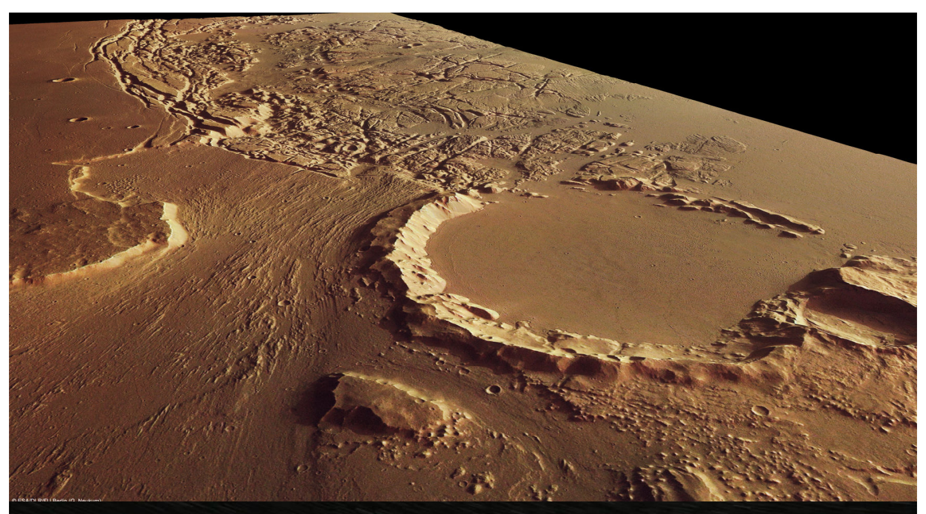
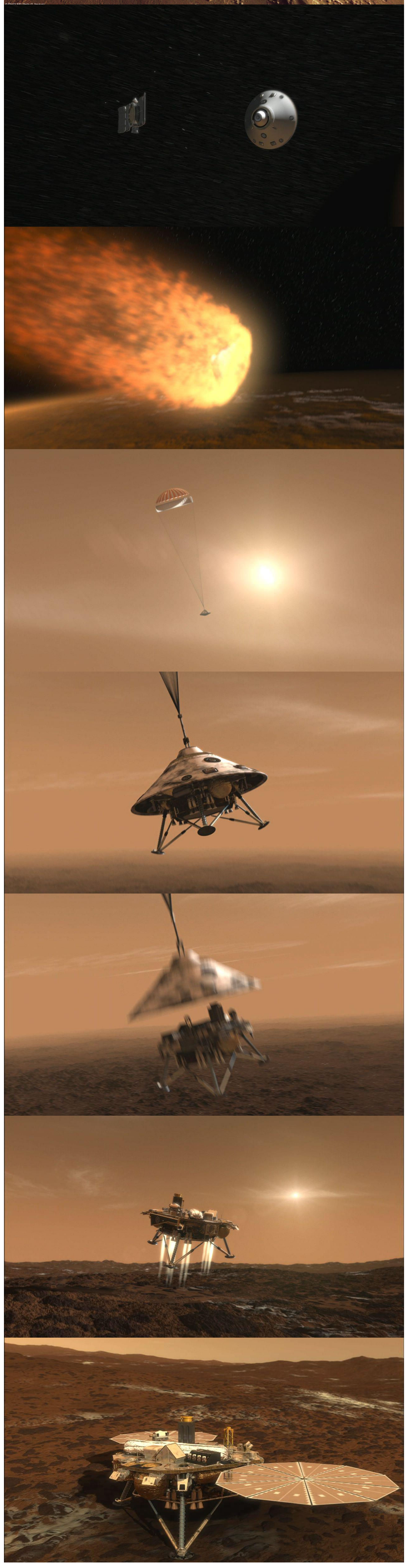


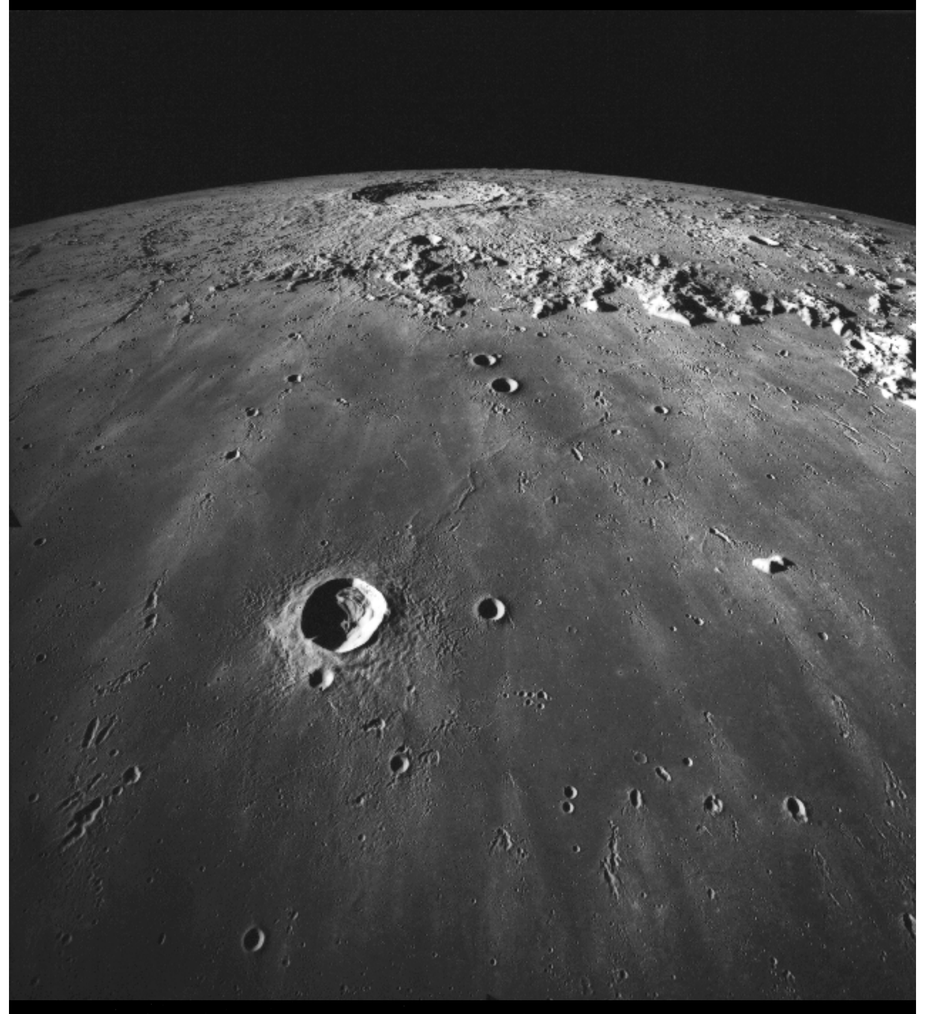
Image © ESA/DLR/FU Berlin - G. Neukum



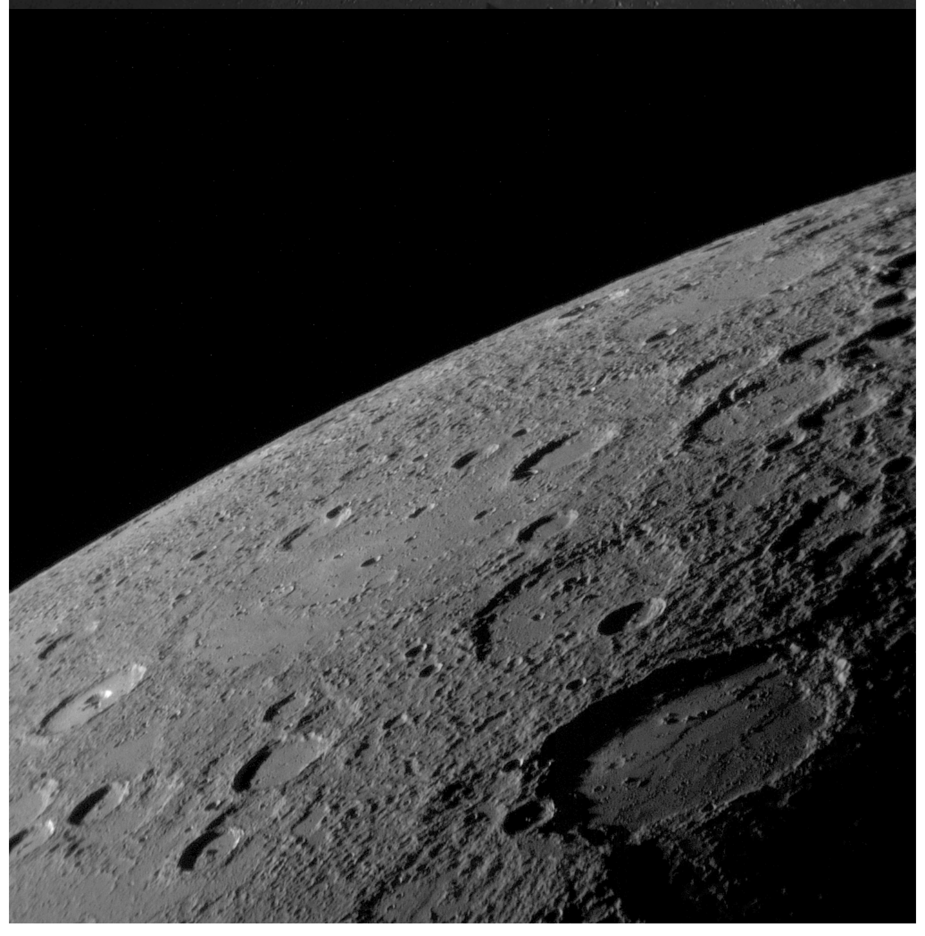
Images courtesy NASA, JPL-Caltech, University of Arizona

No Aerocapture is Possible with Some Targets!

Ida & Dactyl: Image courtesy NASA, JPL



Copernicus Crater: Image courtesy NASA



Mercury: Image courtesy Messenger Teams, JHU APL, NASA

- Goal: A common approach for performing simulation-based systems engineering for planetary probe E/D&L applications.
- Mission concept evaluation:
  - Spacecraft and lander options,
  - Evaluating budgets,
  - Maximising return on investment.
- Mission candidate selection,
- Requirements development,
- Algorithm verification and validation:
  - GNC,
  - Final Landing Site Selection,
  - Hazard Detection & Avoidance.
- A common tool for all subsystems engineers to work with.
- Not Mars-specific! Should be usable for other E/D&L missions too (Moon, Mercury, asteroids.)

What is EAGLE? – "Entry and Guided Landing Environment":

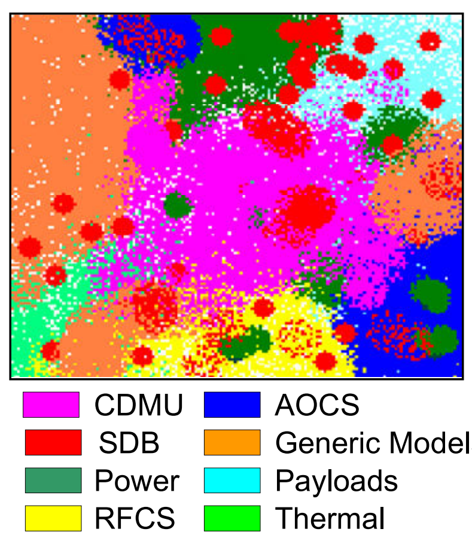
- Simulation framework for multi-body, multi-fidelity simulation.
  - Uses Model Based Design philosophy.
- Who is for?
  - Developed for ESA and aimed at end-end cradle to grave developments:
    - Phase A through to Phase D (and E),
    - Entry to Landing (and Return).
- Why is different?
  - No more single simulator developments. Standards based, extensible, highly configurable.
  - Full lifecycle simulation.

EAGLE: Target Users

- GNC Engineers: Algorithms prototyping, technology maturation, hardware testing.
- System Engineers / Designers: Equipment choices, trade-off studies, performance analysis.
- Mission Engineers / Designers: Mission concept, multiple system interactions, task optimisation.
- Test and Analysis Teams: Validation and verification.
- Allows users with domain specific knowledge to focus on specialist area.
- Provides an integration and test environment for the model, system and mission levels.
- Faster design iteration and increased productivity!

Why EAGLE? Previous Simulators Have not Been Portable

- **Example of what a simulator software architecture may look like:**



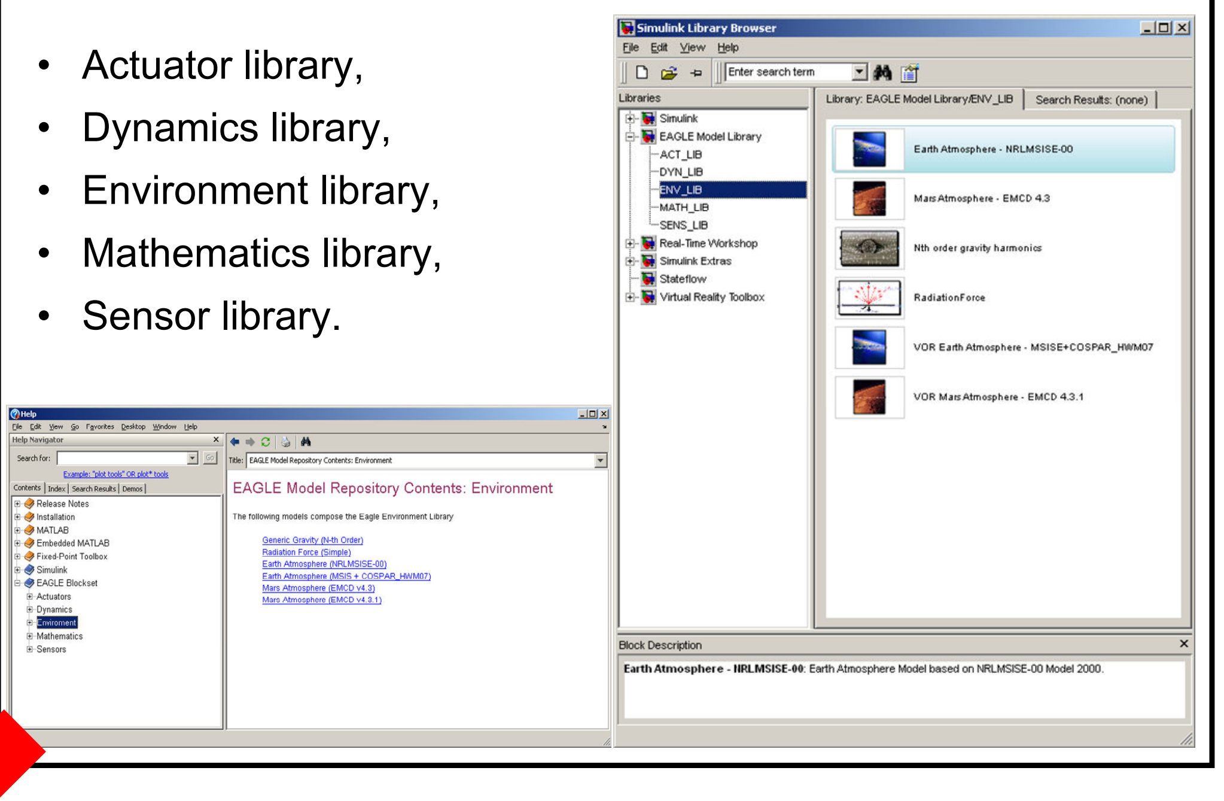
- No clear interface between models.
- Difficult to isolate models for reuse.
- Transforming a working simulator into a new one can be a long and faulty process.
- Software reuse almost impossible.

Situation can be improved:

- Defining clear interfaces between the different elements of the spacecraft.
- Breaking down the whole simulator into models.
- Realizing "plug-and-play" models.

Full MATLAB / Simulink Integration

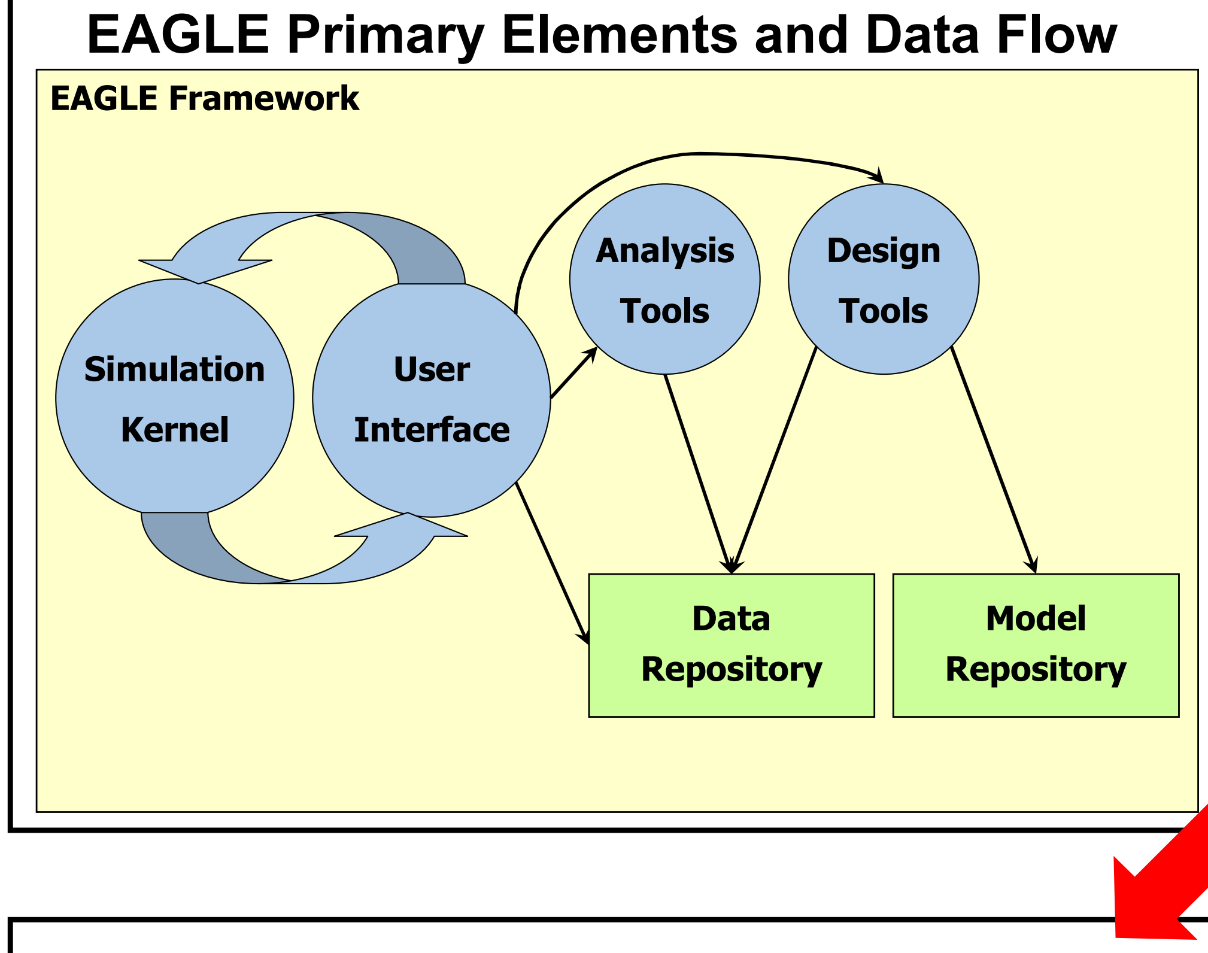
- Actuator library,
- Dynamics library,
- Environment library,
- Mathematics library,
- Sensor library.

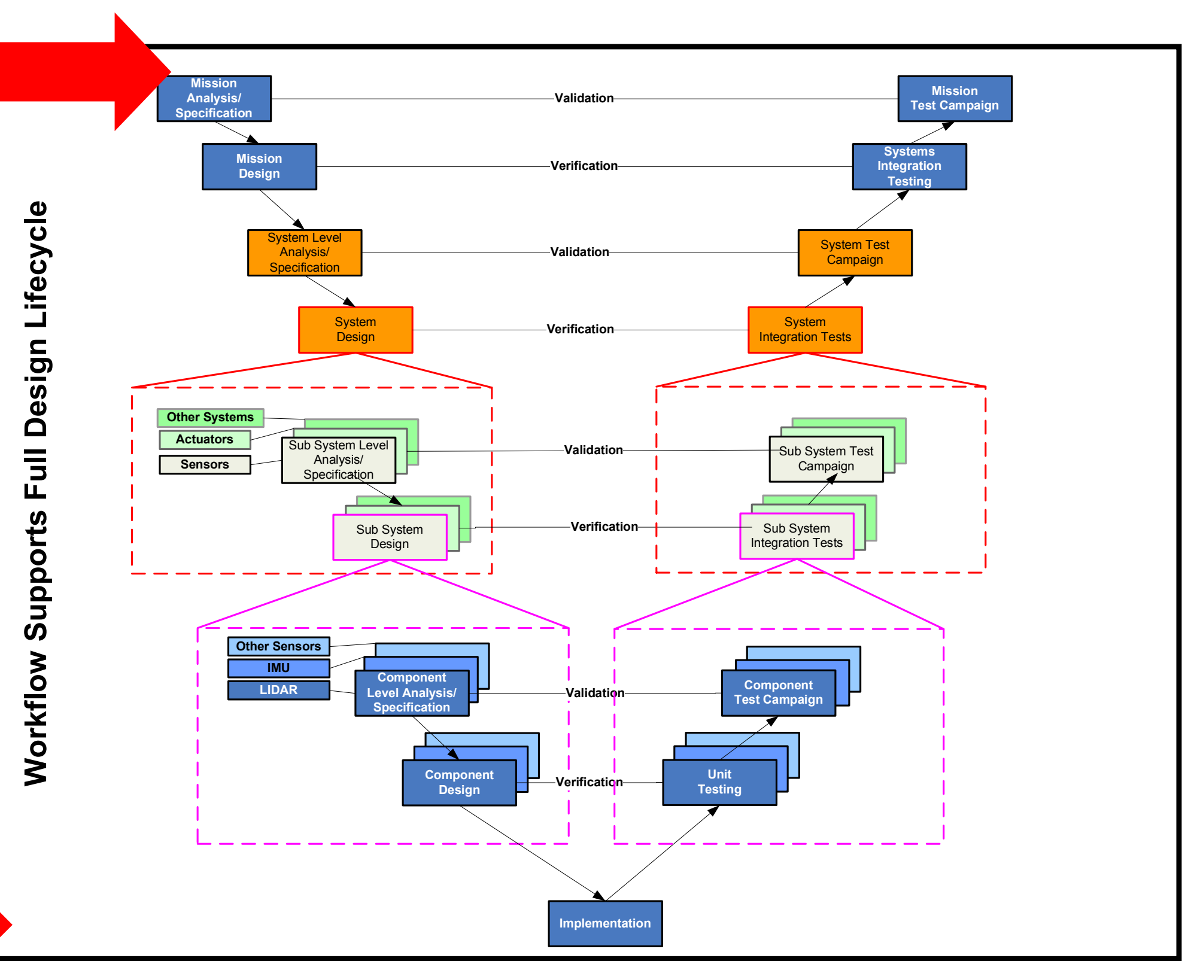


Earth Atmosphere - NRLMSISE-00: Earth Atmosphere Model based on NRLMSISE-00 Model 2000.

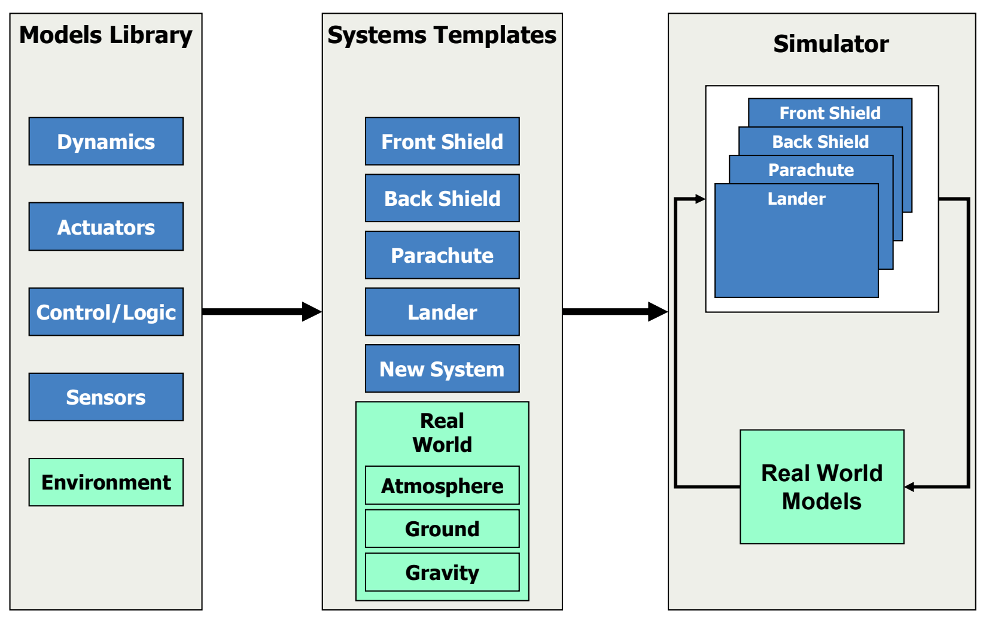
Simulink Library Browser
Help
EAGLE Model Repository Contents: Environment
Radiation Force (Simple)
Earth Atmosphere (NRLMSISE-00)
Earth Atmosphere (MSIS + COSPAR_HWM07)
Mars Atmosphere (EMCD v4.3)
Mars Atmosphere (EMCD v4.3.1)
Fixed-Point Toolbox
Simulink
EAGLE Blockset
Actuators
Dynamics
Enviroment
Mathematics
Sensors
Block Description

EAGLE Primary Elements and Data Flow

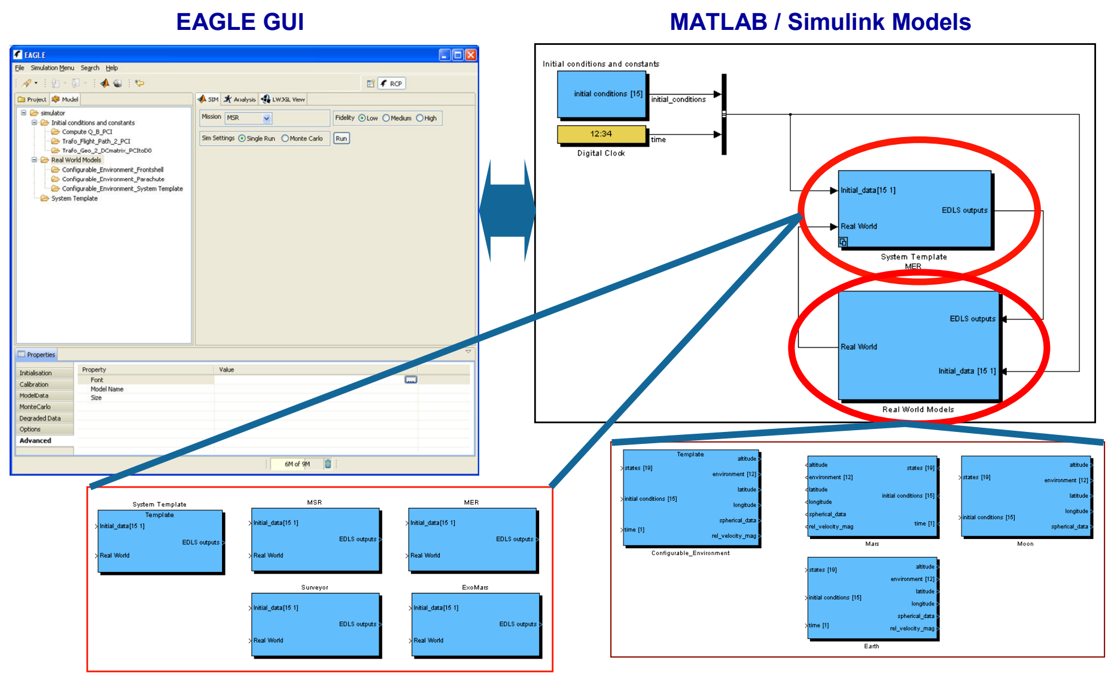




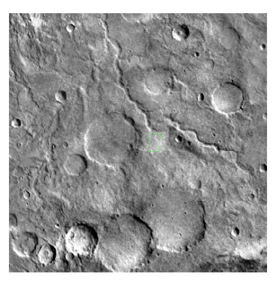
EAGLE Has a Template-Based Architecture



Building a Simulation in EAGLE

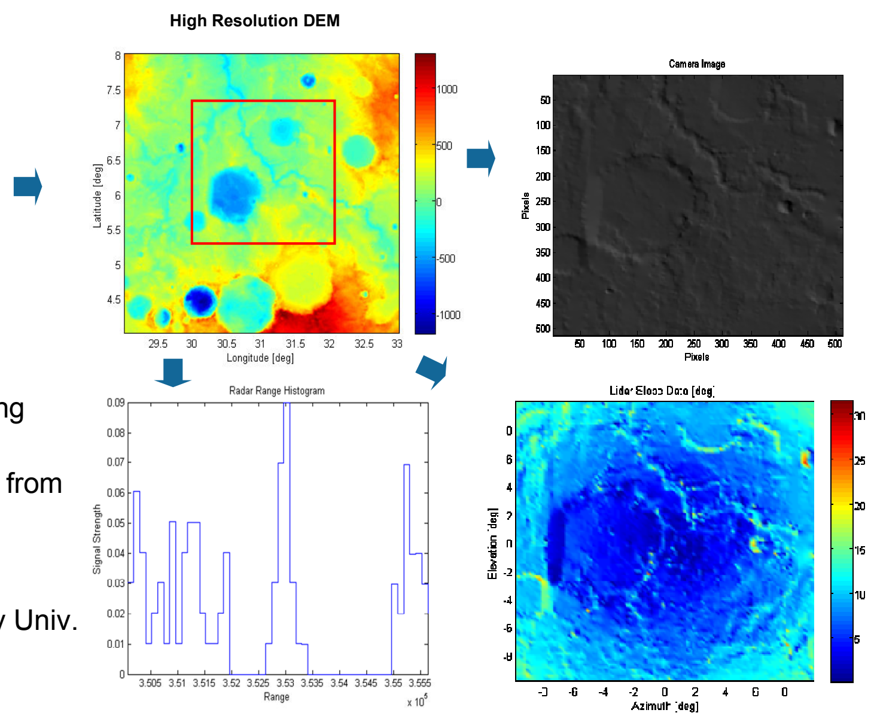


Simulated Vision Sensors: PANGU



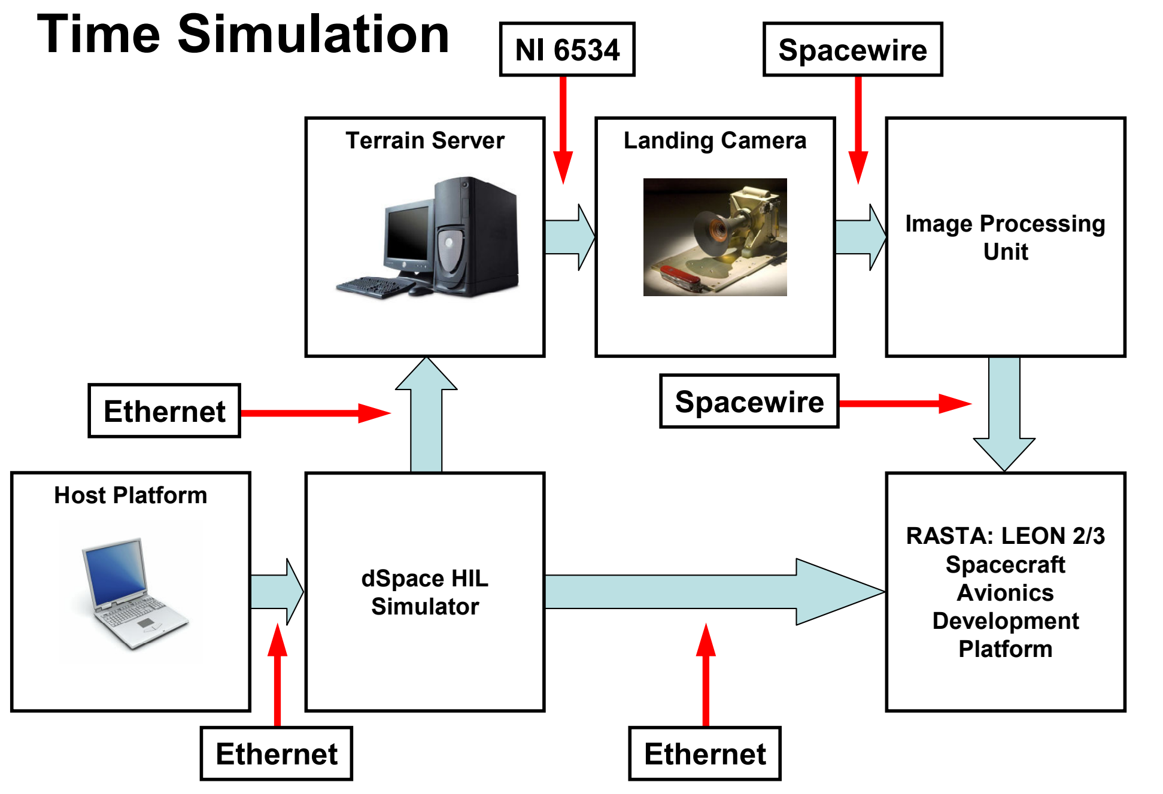
MARS: *NAKTONG VALLIS*

Image courtesy NASA



- Real ground modelling capabilities
- Examples of outputs from various sensors
- Integrates PANGU (Planetary Scene Generation Utility) by Univ. of Dundee.

Avionics Test Bench H/W: Hard Real-Time Simulation



International Planetary Probe Workshop 2010: IPPW-7